

ADAPTIVE PREDICTIVE PLAYOUT SCHEME FOR PACKET VOICE APPLICATIONS

Field of Invention

5 This invention relates to voice packet playout schemes and in particular to an Adaptive Predictive Playout Scheme for Packet Voice Applications.

Background of the Invention

10 The revolution in high-speed communication networks, an example of which is the Internet, has given rise to the potential for enabling the deployment of multimedia applications. These applications, however, require stringent quality of service (QoS) guarantees, such as bounded delay and jitter. The current Internet was originally designed to offer best effort service without any QoS guarantees. In such a packet switching environment, the delay of each packet varies greatly due to the complexities of the network traffic and to the traffic scheduling algorithms implemented for efficient
15 utilization of bandwidth. Voice data or speech packets are generally considered to be transported at a variable bit rate (VBR). As a result, the problem of unbounded jitter, introduced by the networks, often renders the speech unacceptable or even unintelligible. It thus becomes essential to offer control mechanisms to obtain distinctive QoS guarantees.
20

Essentially, voice applications can be broadly classified as either interactive or unidirectional. Serving dissimilar purposes, these two classes of applications differ in playout delay requirements and the tolerances for playout impairment. Interactive voice applications are more sensitive to playout delay than playout impairment due to their real-time nature. It is therefore acceptable in interactive voice applications to trade some
25 playout impairment for better playout delay.

Methods of buffering packets at the receiver end have been extensively studied. Such prior art methods include I-Policy and E-Policy [W.E. Naylor and L. Kleinrock, "Stream Traffic Communication in Packet Switched Networks: Destination Buffering
30 Considerations", IEEE Transactions on Communications, Vol. COM-30, No.12, Dec 1982; and D.L. Stone and K. Jeffay, "An Empirical Study of Delay Jitter Management Policies", Multimedia System, pp.267-279, Vol. 2, No.6, Jan 1995]. However, these schemes do not adapt to traffic conditions, such as delay and jitter, which may vary from time to time. Adaptive playout schemes have also been proposed based on an assumption

that the level of traffic conditions like delay jitter for the near future can be estimated in terms of the observed level in the recent past [D.L. Stone and K. Jeffay, "An Empirical Study of Delay Jitter Management Policies", Multimedia System, pp.267-279, Vol. 2, No.6, Jan 1995].

It is therefore an aspect of an object of the invention to provide a control mechanism for improving the utilization of resources and for optimizing service performance.

Summary of the Invention

According to an aspect of the invention, there is provided an adaptive predictive playout scheme, based on a Least Mean Square (LMS) prediction algorithm, for packet voice applications. The packets are received and stored in a buffer for playout at a constant draining rate P_0 , where P_0 is determined by the codec used.

When the number of packets in the buffer is greater than L_0 , the arrival interval of the next incoming packet is predicted based the LMS prediction algorithm. If the estimated arrival interval for the next packet is smaller than a draining threshold D_0 , then the next packet is predicted to arrive at the destination relatively early and thus is predicted to be buffered relatively longer than the previously received packets. However, if the oldest packet in the buffer is discarded, then the latency (time of packet in the buffer) of the next packet is expected to be reduced, but without increasing the probability of causing a gap as there are a number of packets in buffer queue.

If, however, the prediction for the next packet arrival interval is greater than the draining threshold D_0 , then this next packet is expected to arrive at a time when all of the packets have been played out, thus no packets are needed to be discarded. With such a prediction, the receiver continues to play out the remaining packets provided that the maximum acceptable playout latency is not exceeded. After the playout of the last packet of the talkspurt, or in the event that no packet has arrived for some time since the arrival of the last packet, the talkspurt playout is finished. The receiver starts or resets to playout the next talkspurt.

Brief Description of the Drawings

In the accompanying drawings:

Figure 1 is a time-line diagram illustrating voice source behavior;

Figure 2 is a block diagram illustrating a linear predictor according to the invention;

Figure 3 is a block diagram illustrating an adaptive linear predictor according to the invention;

Figure 4 is a flowchart illustrating LMS prediction algorithms according to the invention;

- 5 Figure 5 is a block diagram illustrating an adaptive prediction payout mechanism utilizing LMS prediction algorithms of Figure 4; and

Figure 6 are flowcharts illustrating an adaptive predictive payout scheme in accordance with Figure 5.

10 Detailed Description of the Preferred Embodiments

For voice data as shown in Figure 1, during a talkspurt of duration $1/\alpha$ packets of speech are generated at fixed intervals T . During silence periods, no packets are generated. At the receiver end, the received constant-size voice packets are played out at a constant bit rate.

- 15 Talkspurts of speech are of relative short duration ($1/\alpha$). At the receiver end, the packet arrival intervals for talkspurts are assumed to be statistically stationary. Consequently, a LMS prediction algorithm can be used to predict the packet arrival intervals.

- Thus, where $x(t)$ ($t=0,1,2,\dots$) denotes a series of packet arrival intervals, the
 20 problem of voice packet arrival interval series prediction involves predicting the value of $x(t+l)$ from the known $x(t-n+1), x(t-n+2), \dots, x(t)$ where $x(t)$ is the most recently received packet. When $l=1$, this process is referred to as *one-step prediction*. The well known least mean square (LMS) error linear prediction is based on Wiener-Hopf equations, whereby a *k-step* linear predictor predicts $x(n+k)$ using a linear combination
 25 of the current and previous values of $x(n)$. Thus, the *p*th-order linear prediction is obtained by the following equation:

$$\hat{x}(n+k) = \sum_{l=0}^{p-1} w(l)x(n-l) \quad (3.1)$$

where $w(l)$ are the prediction filter coefficients, for $l=0,1,2,\dots,p-1$. A linear predictor is illustrated in Figure 2 where

- 30 $w = [w(0), w(1), \dots, w(p-1)]^T$
- $$x(n) = [x(n), x(n-1), \dots, x(n-p+1)]^T$$
- $$e(n) = x(n+k) - \hat{x}(n+k) \quad (3.2)$$

From equations (3.1) and (3.2),

$$e(n) = x(n+k) - \mathbf{w}^T \mathbf{x}(n) \quad (3.3)$$

The optimal linear predictor in the mean square sense is the one that minimizes the mean square error ξ , where

$$\xi = E\{e(n)^2\} \quad (3.4)$$

Since ξ is a quadratic function, it has a unique minimum. Therefore, the vector \mathbf{w} that minimizes ξ is found by taking the gradient of ξ , setting it equal to zero, and then solving for \mathbf{w}

$$\nabla \xi = 0$$

$$\nabla \xi = \nabla E\{e(n)^2\} = -2E\{e(n)\mathbf{x}(n)\} = 0$$

Substituting the value for

$$\nabla \xi = -2E\{[x(n+k) - \mathbf{w}^T \mathbf{x}(n)]\mathbf{x}(n)\} = 0$$

Then

$$E\{x(n+k)\mathbf{x}(n)\} = E\{\mathbf{w}^T \mathbf{x}(n)\mathbf{x}(n)\} \quad (3.5)$$

If $\mathbf{x}(n)$ ($n=0,1,2,\dots$) is wide-sense stationary, the correlation between $\mathbf{x}(n)$ and $\mathbf{x}(n+k)$ is only a function of k , $r_x(k)$.

$$r_x(k) = E\{x(n+k)x(n)\} \quad (3.6)$$

From the left side of equation (3.5),

$$E\{\mathbf{x}(n)\mathbf{x}(n+k)\} = \begin{bmatrix} r_x(k) \\ r_x(k+1) \\ \vdots \\ r_x(k+p-1) \end{bmatrix} = \mathbf{r}(k)$$

From the right side of equation (3.5),

$$E\{\mathbf{w}^T \mathbf{x}(n)\mathbf{x}(n)\} = E\{\mathbf{x}(n)\mathbf{x}(n)^T\} \mathbf{w}^T = \begin{bmatrix} r_x(0) & r_x(1) & \cdots & r_x(p-1) \\ r_x(1) & r_x(0) & \cdots & r_x(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r_x(p-1) & r_x(p-2) & \cdots & r_x(0) \end{bmatrix} \mathbf{w}^T = \mathbf{R}_x \mathbf{w}^T$$

where \mathbf{w} is the vector of coefficients, \mathbf{R}_x is a $p \times p$ Hermitian Toeplitz matrix of auto-correlations, and $\mathbf{r}(k)$ is the vector of cross-correlations between predicted value $x(n+k)$ and $\mathbf{x}(n)$.

Thus,

$$\mathbf{R}_x \mathbf{w}^T = \mathbf{r}(k) \quad (3.7)$$

The equations in (3.7) are the Wiener-Hopf equations for linear prediction. For a one-step prediction ($k=1$), the set of linear equations in (3.6) are equivalent to the set of linear equations used to fit a p th-order autoregressive (AR) process with the exception of a minus sign. The solution to the equations in (3.6) requires knowledge of the auto-correlation of $x(n)$, and it also assumes that $x(n)$ is wide sense stationary, i.e., the mean, variance, and auto-covariance of $x(n)$ do not change with time. It also requires inverting \mathbf{R}_x whose size depends on the order of linear predictor p .

LMS for prediction does not require any prior knowledge of the auto-correlation of a sequence. Therefore, it can be used as an on-line algorithm to predict time intervals. A signal diagram of an adaptive linear predictor is shown in Figure 3. The prediction coefficients $w(n)$ are time-varying. The errors, $\{e(n)\}$ are fed back and used to adapt the filter coefficients in order to decrease the mean square error. As time progresses, a p number of the latest $x(n)$ is captured to predict the value of $x(n+1)$, in the manner of a sliding window over a timeline to predict the next value in terms of a few of the latest values.

The steps of a LMS prediction algorithm according to the invention are: start with an initial estimate of the filter (prediction) coefficients $w(0)$; and for each new data point, compute $\nabla \xi$, where

$$\nabla \xi = -2E\{\tilde{e}(n)x(n)\}.$$

In practice, the statistics are not known and may change with time. Therefore, the expectation operator E is replaced with an estimate. The simplest estimate is the one point sample average $\tilde{e}(n)x(n)$. The $\nabla \xi$ is then used to update $w(n)$ by taking a step of size 0.5μ (μ is an adaptation constant for adjusting the prediction errors) in the negative gradient. The update equations for the LMS filter coefficients are:

$$\begin{aligned} w(n+1) &= w(n) - 0.5\mu \nabla \xi \\ w(n+1) &= w(n) + \mu e(n)x(n) \end{aligned} \quad (3.7b)$$

If $x(n)$ is stationary, $w(n)$ converges to the mean of the optimal solution $\mathbf{R}_x^{-1} \mathbf{W} = \mathbf{r}(k)$.

The LMS thus converges in the mean if $1 < 1/\mu < 2/\lambda_{\max}$, where λ_{\max} is the maximum eigenvalue of \mathbf{R}_x .

According to the invention, a normalized LMS (NLMS) is a modification to the LMS algorithm where the update equation is:

$$w(n+1) = w(n) + \frac{\mu e(n)x(n)}{\|x(n)\|^2} \quad (3.4)$$

where $\|x(n)\|^2 = x(n)^T x(n)$. NLMS has the advantage over LMS of less sensitivity to the

- 5 step size μ . Using a large μ results in a faster convergence and quicker response to signal changes. However, after convergence, the prediction parameters have larger fluctuations. On the other hand, using a small μ results in a slower convergence, but smaller fluctuations after convergence. There is a tradeoff between faster convergence versus smaller fluctuations.

- 10 A flowchart of LMS prediction according to an embodiment of the invention is shown in Figure 4, the steps are: step 400, at the start of a talkspurt, $n=0$, an initial $w(n)$ is estimated; step 405, a packet is received and a packet arrival interval $x(n)$ is obtained; step 410, the next packet arrival interval $x(n+1)$ is predicted or calculated; step 415, another packet is received and the next packet arrival interval $x(n+1)$ is obtained; step 420, the error $e(n)$ is calculated using equation (3.3); step 425, an update coefficient $w(n+1)$ is calculated using equation (3.8) where the Normalized LMS prediction algorithm used; step 430, the LMS prediction algorithm to calculate $x(n)$ is updated with the parameters $w(n+1)$ and $x(n+1)$, and the last interval parameters $w(n-p+1)$ and $x(n-p+1)$ are dropped; and step 435, go to step 410 until the talkspurt ends.

- 20 According to another embodiment as also shown in Figure 4, equation (3.7b) is substituted for equation (3.8) in step 425 where the LMS prediction algorithm is used.

- An adaptive predictive playout mechanism, based on LMS prediction of Figure 4, is shown in Figure 5. It is composed of three components: 1) a smoothing buffer 10. 2) an LMS traffic predictor 12, and 3) a CBR (Constant Bit Rate) player 14. The arriving packets are queued in the smoothing buffer 10. LMS predictor 12 employs an online algorithm as shown in Figure 4, using the normalized LMS prediction algorithm, to predict the arrival interval of next incoming packet. Based on the predicted packet arrival interval, the CBR player 14 derives an adaptive buffer delay by means of discarding the oldest packets in the buffer if necessary.

- 30 The first few packets of each talkspurt are buffered to smooth the jitter. There are two conditions for starting playout of packets: current buffer length Q is greater than the buffer threshold L_0 , and queuing time of the oldest packet in buffer B is greater than the

maximum acceptable playout latency T_0 . Whenever either of these two conditions is met, the CBR player 14 starts playout of packets at a constant bit rate.

During the playout of the packets at a constant draining rate P_0 , where P_0 is determined by the codec used to encode the talkspurt into the packets, when the number of packets in the buffer is greater than L_0 , the arrival interval of the next incoming packet is predicted by the LMS predictor 12. If the estimated next arrival interval is smaller than the draining threshold D_0 , then this packet is predicted to arrive at the destination relatively early and that this packet is predicted to be buffered relatively longer than the previously received packets. If the oldest packet in the buffer is discarded, the latency of the next incoming packet is expected to be reduced, but without increasing the probability of causing a gap as there are a number of packets in buffer queue. If the prediction of the packet arrival interval is greater than the draining threshold D_0 , then this packet is expected to arrive at a time when all the packets have been played out, thus no packet needs to be discarded. With such a prediction, the receiver continues to play out the remaining packets provided that the maximum acceptable playout latency is not exceeded. After the playout of the last packet of the talkspurt, or no packet has arrived for some time since the arrival of the last packet, the talkspurt playout is finished. The receiver starts or resets to playout the next talkspurt.

Flowcharts of the operation of the adaptive predictive playout mechanism of Figure 5 are shown in Figure 6. The parameters B , T_0 , D_0 , L_0 , and Q , for the mechanism are also shown. The steps of the operation are: step 600 waiting for a talkspurt; step 610, receipt of a new talkspurt; step 620, initial smoothing of the packets of the talkspurt, which comprises receiving packets 622 and holding the packets in the smoothing buffer 10 until the current buffer length Q reaches the threshold L_0 or B (= Queuing time of the oldest packet in buffer) is greater than T_0 (= Maximum acceptable playout latency) 624; and playout 640 of the packets in the buffer 10.

The playout 640 comprises step 642 to playout the oldest packet in buffer 10 with a constant draining rate P_0 as determined by the codec used to encode the packets; step 644, the buffer length Q is checked to determine if the last packet in the buffer 10 has been playout and if played out then go to step 600 to wait for the next talkspurt; if not played out then go to step 642 to playout the next packet.

As the packets are being played out, further packets are also being received and added to the buffer 10 (step 646). For each received packet, the LMS predictor 12 is updated accordingly to the normalized LMS prediction algorithm (step 648) and the buffer

length Q is checked to determine if Q is below the buffer threshold $L0$ (step 650). If Q is greater than $L0$ then predict a next incoming packet arrival interval d (step 652). The interval d is compared (step 654) with the draining threshold $D0$ to control possible flooding where d is not greater than $D0$, and also to insure that the maximum playout latency $T0$ still remains acceptable where B is not greater than $T0$. If either of the conditions in step 654 is not satisfied then discard the oldest packet in the buffer 10 (step 656).

Various simulation scenarios have been tested using simulations of the adaptive prediction playout scheme of the invention. Without limiting the scope of the invention, the results of the estimated probabilistic QoS values (delay, delay jitter, loss and gap probabilities) within the range of specified operating parameter values are provided herein below. With parameter values specified as follows :

- Exponential packet arrival with the mean varying between 1.5 ms and 3ms,
- Buffer threshold $L0$ of 50, 55, 60, 75 or 100 packets,
- Maximum acceptable playout latency $T0$ of 150ms,
- Draining threshold $D0$ of 6ms,
- Constant packet draining rate $P0$ of 1 packet every 1.5 ms or 3 ms,
- Packet length of 1024 bits,
- Prediction step size μ of 0.05, and
- Sliding window size p of 1, 3, 5 or 10;

it was observed that,

- As the sliding window size increased, the packet gap or lost probabilities (varying between 0.3% and 1.1%) increased, with the most drastic deterioration occurring when the window size jumped from 1 to 3; increasing the buffer threshold decreased the values of gap or lost probabilities, which are annoying to voice users when they are too high;
- As the buffer threshold increased, the mean of queuing delay (varying between 80 ms and 148 ms) also increased proportionally; decreasing window size improved the delay with a very strong improvement occurring when window size was reduced from 3 to 1; and
- Delay jitter statistics were not collected in the initial experimentation due to the fact that their impact on voice QoS is accounted for by the packet lost or gap probabilities, and the packet draining rate (which was constant).

Although preferred embodiments of the invention have been described herein, it will be understood by those skilled in the art that variations may be made thereto without departing from the scope of the invention or the scope of the appended claims.